Global Learning of Focused Entailment Graphs

Jonathan Berant, Ido Dagan and Jacob Goldberger

ISCOL 2010*

16/6/10

*Accepted as full paper to ACL 2010

Outline

- Textual entailment (TE)
- Background: learning entailment rules for predicates
- Entailment graph structure
- Application of entailment graph
- Global algorithm for learning entailment graphs
- Results

Textual Entailment (TE)

- A directional relation between two text fragments:

A text *‘t’* entails a hypothesis *‘h’* (denoted by *t*→*h*) if humans reading *‘t’* infer that *‘h’* is most likely true.

- Useful for applications:
  - Machine Translation
  - Information Retrieval
  - Information Extraction

Knowledge Resources for TE

- TE systems employ knowledge resources that contain entailment rules.
- <u>Entailment rules for predicates</u>: contain predicates and arguments.

X is a symptom of Y → Y cause X

- We focus on learning such entailment rules.

Local Learning

- Manually-prepared resources (Szpektor and Dagan, 2009):
  - WordNet relations: hyponym, derivation
- Pattern-based methods (Chklovsky and Pantel, 2004):
  - "he scared and even startled me" → *X startle Y → X scare Y*
- Distributional similarity: distribution of arguments predicts similarity between predicates.

Distributional Similarity

X affect Y		
X	Y	#
insulin	metabolism	7
Zantac	BP	4
…	…	…

X treat Y		
X	Y	#
Zantac	BP	9
diet	diabetes	2
…	…	…

- Yates and Etzioni (2009) estimate the probability that two predicates are synonymous using a generative model.

$$Lin(u,v) = \frac{\sum_{f \in F_x^u \cap F_x^v} [w_x^u(f) + w_x^v(f)]}{\sum_{f \in F_x^u} w_x^u(f) + \sum_{f \in F_x^v} w_x^v(f)}$$

$$cover(u,v) = \frac{\sum_{f \in F^u \cap F^v} w^u(f)}{\sum_{f \in F^u} w^u(f)}$$

$$DIRT(u,v) = \sqrt{Lin_x(u,v) \bullet Lin_y(u,v)}$$

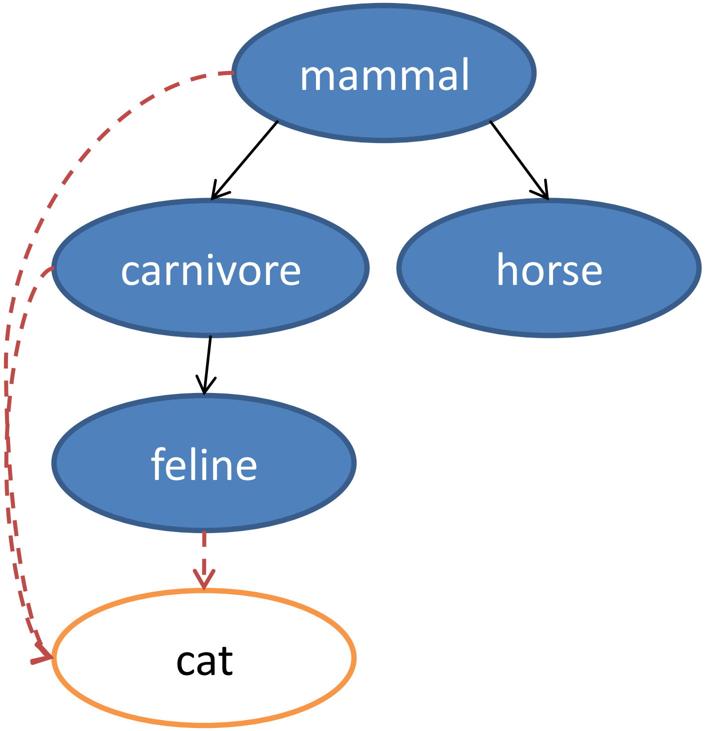
$$Binc(u,v) = \sqrt{Lin(u,v) \bullet cover(u,v)}$$

(Lin and Pantel, 2001)

(Szpektor and Dagan, 2008)

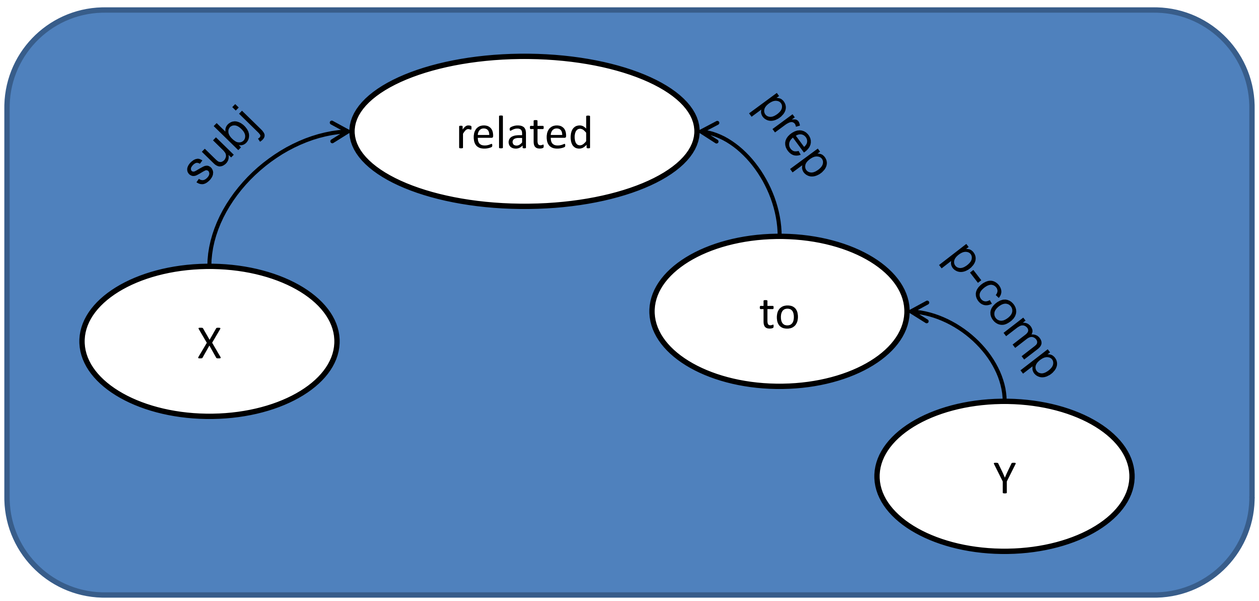
Global Learning

- Snow et al. (2006) presented an algorithm for taxonomy induction.
- At each step they add the concept that maximizes the likelihood of the taxonomy given the transitivity constraint.



Propositional Templates

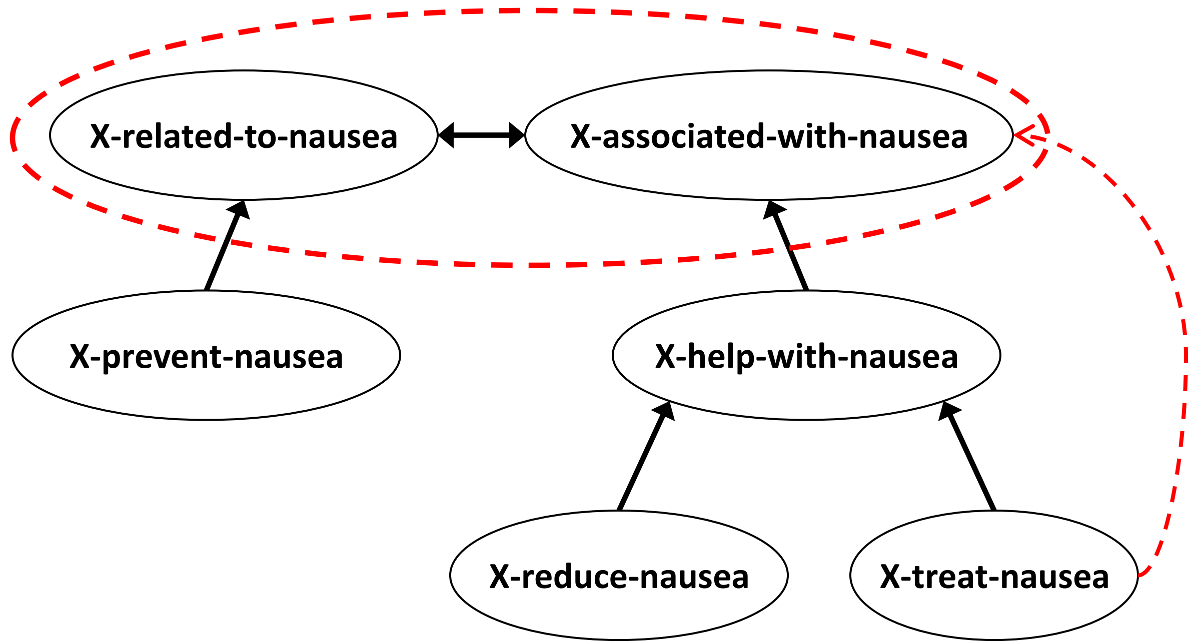
- Dependency path containing a predicate and two arguments (possibly one is instantiated)



Entailment Graph

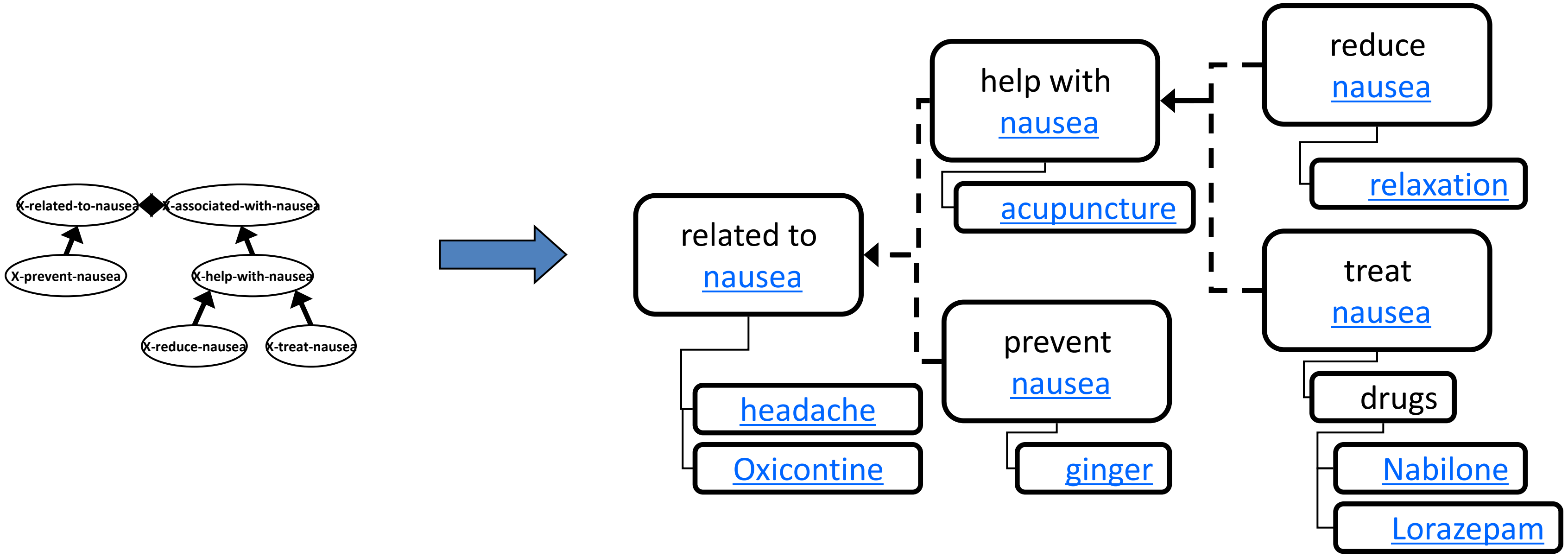
- Nodes: propositional templates
- Edges: entailment between templates
- Assumption: Predicates are monosomous
  - Small graphs
  - One topic

Entailment Graph Properties



- Edges are transitive (monosemous predicates).
- Strong connectivity components represent synonyms.
- Merging strong connectivity components to a single node results in a Directed Acyclic Graph.

Hierarchical Summarization



- Scenario: user queries about a concept *(nausea)* and would like a structural output.
- Summarize the propositions of the corpus using a predicate entailment hierarchy interleaved with a taxonomy.

Learning Entailment Graph Edges

- Two step algorithm:
  1. Train a **local** entailment classifier **once**: given a pair of propositional templates (t_1,t_2), estimate whether $t_1 \rightarrow t_2$
  2. Given the nodes of an entailment graph, learn the edges of the graph using the entailment classifier

Entailment Classifier - Outline

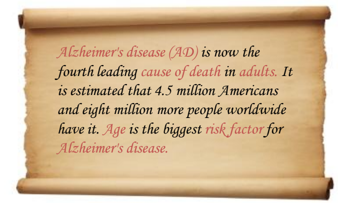
- Input: large corpus and lexical database
- Steps:

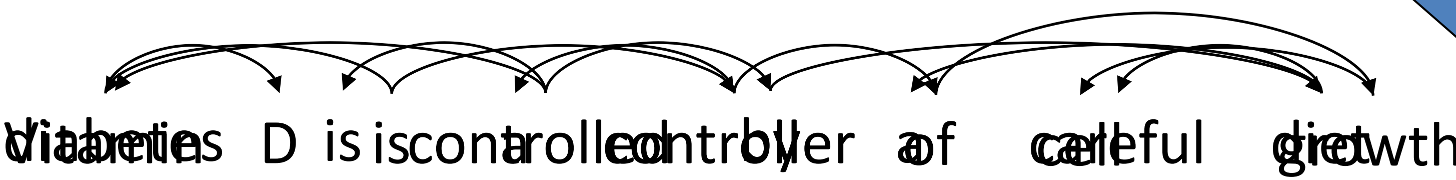
1. Extract propositional templates from corpus
2. Generate automatically positive and negative examples using lexical database
3. Represent train set using distributional similarity

- Output: local entailment classifier

Template Extraction

1. Parse a large corpus
2. Extract and normalize tuples (á la Etzioni)
3. Replace arguments with variables

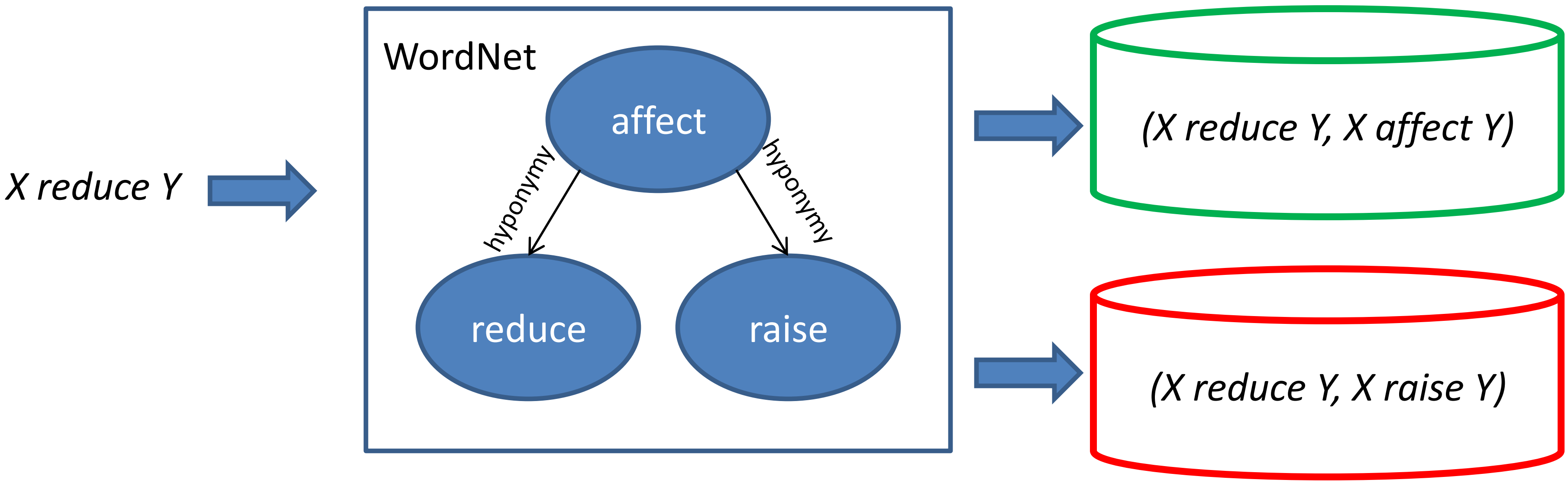




1. X control Y
2. …
3. …

Train Set Generation

- Use propositional templates from the corpus and lexical database:



- Generation method similar to “distant supervision” (Snow et al., 2005).

Representing Template Pairs

- A pair *(t_1,t_2)* is represented by various distributional similarity algorithms.

Measure	score
DIRT	0.549
BINC	0.919
TEASE	0.711
…	0.0

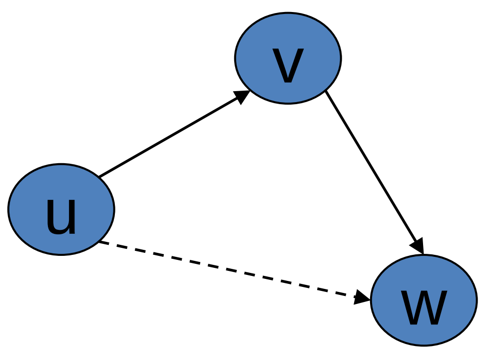
- Reminiscent of the verb disambiguation algorithm proposed by Connor and Roth (2007)

Global Learning of Edges

- Learn the edges E of over a set of nodes V using Integer Linear Programming:
  - Binary variables I_{uv} for every pair of nodes
  - Global transitivity constraint
  - Target function maximizes scores of edges in the graph
- Problem is NP-hard by a reduction from “Transitive Graph” (Yannakakis, 1978).

Global Learning of Edges - constraints

- Initial information: a set *Pos* of node pairs that are edges and a set *Neg* of node pairs that are not edges.
- Constraints:



$$\forall u,v,w \in V\ I_{uv} + I_{vw} - I_{uw} \leq 1$$

$$\forall (u,v) \in Neg\ I_{uv} = 0$$

$$\forall (u,v) \in Pos\ I_{uv} = 1$$

Target Function

- Let F_{uv} be the features for the node pair *(u,v)* and F be the union over all node pairs.
- Given a probabilistic classifier that estimates $P_{uv} = P(I_{uv}=1|F_{uv})$ we can show that:

$$\hat{G} = \arg\max P(G|F)$$

$$= \arg\max \sum_{u \neq v} \log \frac{P_{uv} \cdot P(I_{uv}=1)}{(1-P_{uv}) \cdot P(I_{uv}=0)} \cdot I_{uv}$$

$$= \arg\max\left(\sum_{u \neq v} \log \frac{P_{uv}}{(1-P_{uv})} \cdot I_{uv}\right) + \lambda \cdot |E|$$

Experimental Evaluation

- 50 million word tokens healthcare corpus
- Ten medical students prepared gold standard graphs for 23 medical concepts:
  - Smoking, seizure, headache, lungs, diarrhea, chemotherapy, HPV, Salmonella, Asthma, etc.
- Evaluation:
  - F_1 on set of edges
  - F_1 on set of propositions

Evaluated algorithms

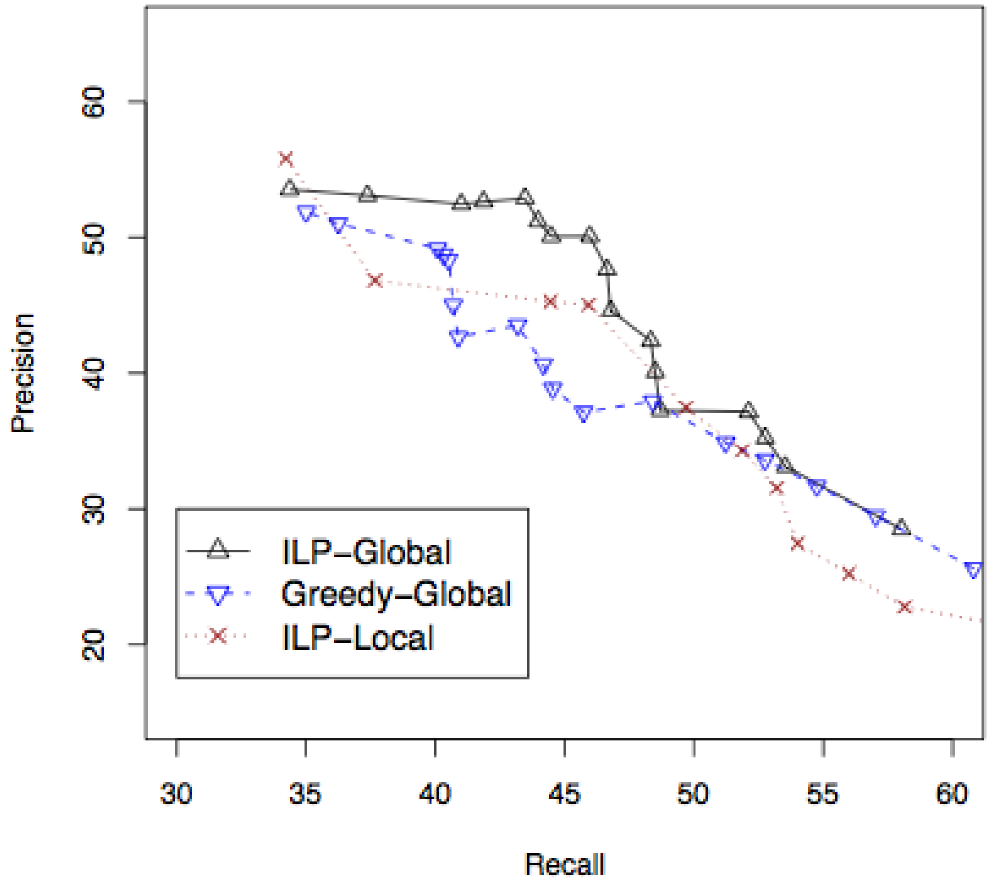
- Local algorithms
  - Single distributional similarity
  - WordNet
  - ILP with No transitivity constraints
- Global algorithms
  - Linear programming/greedy optimization (Snow)

Results

	Edges			Propositions		
	recall	Precision	F_1	recall	Precision	F_1
ILP-global	46.0	50.1	**43.8***	67.3	69.6	**66.2***
Greedy	45.7	37.1	36.6	64.2	57.2	56.3
ILP-local	44.5	45.3	38.1	65.2	61.0	58.6
Local_1	53.5	34.9	37.5	73.5	50.6	56.1
Local_2	52.5	31.6	37.7	69.8	50.0	57.1
Local*_1	53.5	38.0	39.8	73.5	54.6	59.1
Local*_2	52.5	32.1	38.1	69.8	50.6	57.4
WordNet	10.8	44.1	13.2	39.9	72.4	47.3

- The algorithm significantly outperforms all other baselines.

Recall-Precision Curve

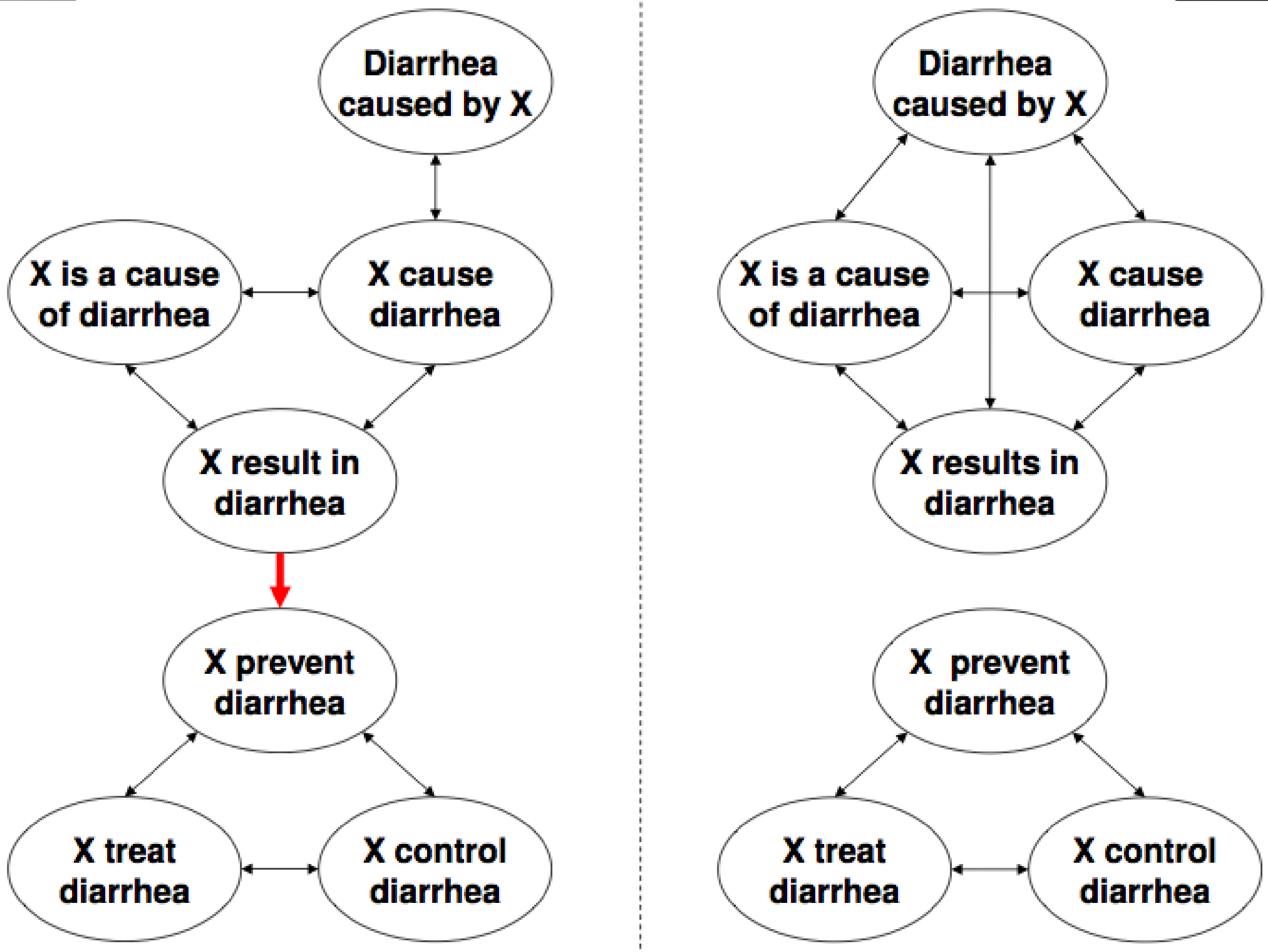


Results

	Global=true/Local=false	Global=false/Local=true
Gold standard = true	48	42
Gold standard = false	78	494

- Comparing disagreement between best local and global algorithms reveals that the global algorithms avoids many false positives.

Local | **Global**



Conclusions

- Algorithm for learning entailment graphs using transitivity and ILP.
- Algorithm significantly outperforms local methods and greedy global methods (Snow et al.).
- Future work: Scale algorithm to handle larger graphs.

Thank you!